Private linear programming without constraint violations

Andrés Muñoz
Google Research
ammedina@google.com

Umar Syed
Google Research
usyed@google.com

Sergei Vassilvitskii
Google Research
sergeiv@google.com

Ellen Vitercik
Carnegie Mellon University
vitercik@cs.cmu.edu

Abstract

We show how to solve linear programs whose constraints depend on private data. Existing techniques allow constraint violations whose magnitude is bounded in terms of the differential privacy parameters ϵ and δ. In many applications, however, the constraints cannot be violated under any circumstances. We demonstrate that straightforward applications of common differential privacy tools, such as Laplace noise and the exponential mechanism, are inadequate for guaranteeing that the constraints are satisfied. We then present a new differentially private mechanism that takes as input a linear program and releases a solution that satisfies the constraints and differs from the optimal solution by only a small amount. Empirically, we show that alternative mechanisms do violate constraints in practice.

1 Introduction

Linear programming (LP) is a fundamental tool in computer science. A diverse array of problems can be formulated as linear programs, including those from fields such as machine learning, engineering, manufacturing, and transportation. The past several decades have seen the development of a variety of linear programming algorithms with provable guarantees, as well as fast commercial solvers.

The goal in linear programming is to find a vector $\boldsymbol{x}$ maximizing an objective function $\boldsymbol{c}^\top \boldsymbol{x}$ subject to the constraint that $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}$. The LP formulation encodes data about the specific problem at hand. In many applications, such as those from the medical domain, this data is defined by individuals' private information. Releasing the LP's solution would thereby leak information about this sensitive data.

As a concrete example, suppose there is a hospital with branches located throughout a state, each of which has a number of patients with a certain disease. Each branch requires a specific drug to treat these patients, which it can obtain from a number of different pharmacies. There is a cost to transporting the drug from any one pharmacy to any one hospital branch. The goal is to determine which pharmacies should supply which hospitals, such that the total cost is minimized. In Figure 3 in Appendix A, we present the LP formulation of this problem. The LP is defined by sensitive information: the constraints reveal the number of diseased patients at each branch.

We provide tools with provable guarantees for solving linear programs while preserving *differential privacy* (DP) [4]. This problem falls in the category of private optimization, for which there are multiple algorithms [1, 2, 7] in the unconstrained case. To our knowledge, only Hsu et al. [6] study differentially private linear programming — by definition, a constrained optimization problem. They allow their algorithm's output to violate the constraints, which can be unacceptable in certain applications. In our transportation example, if the constraints are violated, then some hospital will not

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

receive the drugs they require, or some pharmacy will be asked to supply more drugs than they have in their inventory. The importance of satisfying constraints motivates this paper's central question: how can we privately solve linear programs while ensuring that no constraint is violated?

Our contributions. Formally, our goal is to privately solve linear programs of the form

$$\max_{\boldsymbol{x}\in\mathbb{R}^n} \left\{\boldsymbol{c}^\top \boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\right\}, \tag{1}$$

where $\boldsymbol{b}(D)$ depends on a private database D. Each database is a set of individuals' records, each of which is an element of a domain $\mathcal{X}$. Our algorithm privately maps $\boldsymbol{b}(D)$ to a nearby vector $\bar{\boldsymbol{b}}(D)$ and releases the vector maximizing $\boldsymbol{c}^\top \boldsymbol{x}$ such that $\mathbf{A}\boldsymbol{x} \leq \bar{\boldsymbol{b}}(D)$. We ensure that $\bar{\boldsymbol{b}}(D) \leq \boldsymbol{b}(D)$, and therefore our algorithm's output satisfies the constraints $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)$. This requirement precludes our use of traditional DP mechanisms: perturbing each component of $\boldsymbol{b}(D)$ using the Laplace, Gaussian, or exponential mechanisms would not result in a vector that is component-wise smaller than $\boldsymbol{b}(D)$.

We prove that if $\boldsymbol{x}(D)$ is the vector our algorithm outputs and $\boldsymbol{x}^*$ is the optimal solution to the original LP (Equation (1)), then $\boldsymbol{c}^\top \boldsymbol{x}(D)$ is close to $\boldsymbol{c}^\top \boldsymbol{x}^*$. Our bound depends on the standard differential privacy parameters ϵ and δ. It also depends on the sensitivity Δ of the private LPs, where Δ is the maximum ℓ_∞-norm between any two vectors $\boldsymbol{b}(D)$ and $\boldsymbol{b}(D')$ when D and D' are *neighboring*, in the sense that D and D' differ on at most one individual's data. Finally, our bound depends on the "niceness" of the matrix $\mathbf{A}$, which we quantify using the *condition number* $\gamma(\mathbf{A})$ of the LP [8, 9]. We prove that $\left|\boldsymbol{c}^\top \boldsymbol{x}^* - \boldsymbol{c}^\top \boldsymbol{x}(D)\right| = O\left(\left(\frac{1}{\epsilon}+1\right)||\boldsymbol{c}||_2 \gamma(\mathbf{A})\Delta \ln\frac{1}{\delta}\right)$.

2 Differential privacy definition

To define differential privacy (DP), we first introduce the notion of a neighboring database: two databases $D, D' \subseteq \mathcal{X}$ are *neighboring*, denoted $D \sim D'$, if they differ on any one record ($|D \,\Delta\, D'| = 1$). We use the notation $\boldsymbol{x}(D) \in \mathbb{R}^n$ to denote the random variable corresponding to the vector that our algorithm releases (non-trivial DP algorithms are, by necessity, randomized). Given privacy parameters $\epsilon > 0$ and $\delta \in (0, 1)$, the algorithm satisfies (ϵ, δ)-*differential privacy* ((ϵ, δ)-**DP**) if for any neighboring databases D, D' and any subset $V \subseteq \mathbb{R}^n$, $\Pr[\boldsymbol{x}(D) \in V] \leq e^\epsilon \Pr[\boldsymbol{x}(D') \in V] + \delta$.

3 Single-dimensional linear programs

We begin by analyzing a family of simple, single-dimensional LPs: $\max_{x\in\mathbb{R}} \{x : x \leq b(D)\}$, where $b(D)$ is a real value. This analysis serves as a building block for our analysis of general, multi-dimensional linear programs. Without a privacy requirement, the solution to this LP is trivially $b(D)$. Our goal is to release a private version of $b(D)$, which should be at most $b(D)$ but as close to $b(D)$ as possible. We use the notation Δ to denote the sensitivity of $b(D)$: $\Delta = \max_{D\sim D'} |b(D) - b(D')|$.

A tempting first approach might be to use a variation on the Laplace mechanism [4]: draw $\eta \sim \text{Laplace}\left(\frac{\Delta}{\epsilon}\right)$ and release $x(D) = b(D) - |\eta| < b(D)$. This approach, however, introduces an undesirable dependency between ϵ and δ. For example, suppose $\Delta = 1$. Let $D \sim D'$ be two databases with $b(D) = b(D') + 1$. The probability mass of $x(D')$ for all values larger than $b(D')$ is zero. Meanwhile, the probability mass of $x(D)$ between $b(D')$ and $b(D)$ is $1 - \exp(-\epsilon)$. Therefore, this approach requires that δ be at least $1 - \exp(-\epsilon)$, which can be too large to provide any reasonable privacy guarantees. The exponential mechanism [10] also fails as it cannot ensure that $x(D) \leq b(D)$.

As our attempt at using the Laplace mechanism reveals, we must choose the distribution for $x(D)$ in such a way that the probability mass between $b(D) - \Delta$ and $b(D)$ is at most δ. This is because for any neighboring database D' with $b(D') \leq b(D)$, the probability mass of $x(D')$ in $[b(D'), b(D)]$ must be zero, which forces the probability mass of $x(D)$ in $[b(D'), b(D)]$ to be at most δ.

Letting $s = \frac{\Delta}{\epsilon}\ln\left(\frac{e^\epsilon - 1}{2\delta} + 1\right) = O\left(\Delta + \frac{\Delta}{\epsilon}\ln\frac{1}{\delta}\right)$, we define the density function of $x(D)$ as

$$f_D(u) = \begin{cases} \frac{1}{Z(D)} \exp\left(-\frac{\epsilon|u+s-b(D)|}{\Delta}\right) & \text{if } u \in [b(D) - 2s, b(D)] \\ 0 & \text{else.} \end{cases}$$

where $Z(D) = \int_{b(D)-2s}^{b(D)} \exp\left(-\frac{\epsilon|u+s-b(D)|}{\Delta}\right) du$ is a normalizing constant. We illustrate f_D in Figure 1. Fact 3.1 summarizes a quality guarantee which follows from the support of $x(D)$.

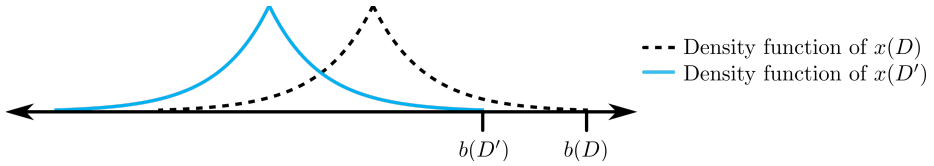


Figure 1: Densities of $x(D)$ and $x(D')$ for two neighboring databases D and D', illustrated over their supports. To preserve DP, we ensure that the probability $x(D)$ is in the interval $[b(D'), b(D)]$ is at most δ, since this interval is disjoint from the support of $x(D')$.

Fact 3.1. *For any database D, with probability 1, $b(D) - O\left(\Delta + \frac{\Delta}{\epsilon} \ln \frac{1}{\delta}\right) \leq x(D) \leq b(D)$.*

The full proof of the following privacy guarantee is in Appendix C.

Theorem 3.2. *Our algorithm preserves (ϵ, δ)-DP.*

Proof sketch. Let V be an arbitrary subset of $\mathbb{R}$ and let D and D' be two neighboring databases. Without loss of generality, suppose $b(D') \leq b(D)$, as in Figure 1. We decompose V into two subsets, $V \cap (-\infty, b(D')]$ and $V \cap (b(D'), \infty)$. We use the exponential decay of the density function f_D to show that $\Pr[x(D) \in V \cap (-\infty, b(D')]] \leq e^{\epsilon} \Pr[x(D') \in V]$. Next, we show that our careful choice of s ensures that $\Pr[x(D) \in V \cap (b(D'), \infty)] \leq \delta$. Therefore, $\Pr[x(D) \in V] \leq e^{\epsilon} \Pr[x(D') \in V] + \delta$. By a similar argument, $\Pr[x(D') \in V] \leq e^{\epsilon} \Pr[x(D) \in V] + \delta$, so the theorem holds. □

4 Multi-dimensional linear programs

We now move to privately solving multi-dimensional linear programs of the form $\max_{\boldsymbol{x} \in \mathbb{R}^n} \left\{\boldsymbol{c}^\top \boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\right\}$, where $\boldsymbol{b}(D) = (b(D)_1, \ldots, b(D)_m) \in \mathbb{R}^m$. Preserving DP while ensuring the constraints are always satisfied is impossible if the feasible regions change drastically across databases. For example, if D and D' are neighboring databases inducing disjoint feasible regions, there is no (ϵ, δ)-DP mechanism that always satisfies the constraints with $\delta < 1$ (see Proposition C.1 in Appendix C). To circumvent this impossibility, we assume that the intersection of the feasible regions across all databases is nonempty. For example, if the origin is always feasible, this assumption is satisfied.

Assumption 4.1. *The set $S^* := \bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$ is non-empty.*

In our approach, we map each vector $\boldsymbol{b}(D)$ to a random variable $\bar{\boldsymbol{b}}(D) \in \mathbb{R}^m$ and release

$$\boldsymbol{x}(D) := \text{argmax}_{\boldsymbol{x} \in \mathbb{R}^n} \left\{\boldsymbol{c}^\top \boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \bar{\boldsymbol{b}}(D)\right\}. \tag{2}$$

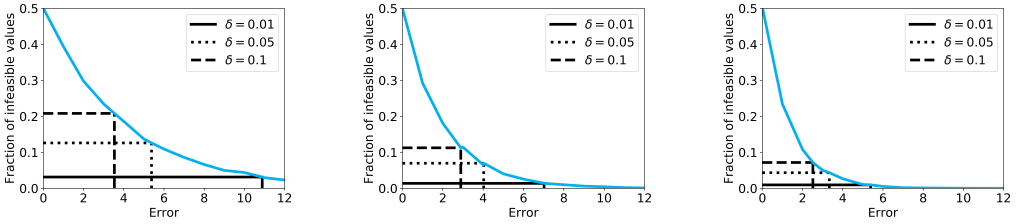
In essence, each component of $\bar{\boldsymbol{b}}(D)$ is defined by the same distribution we used to perturb the one-dimensional constraint in Section 3. We must ensure, however, that the resulting LP is feasible.

To formally describe our approach, we use the notation $\Delta = \max_{D \sim D'} ||\boldsymbol{b}(D) - \boldsymbol{b}(D')||_\infty$ to denote the constraint's sensitivity. We define $\bar{\boldsymbol{b}}(D)_i = \max\left\{\tilde{\boldsymbol{b}}(D)_i, b_i^*\right\}$, where $\tilde{\boldsymbol{b}}(D)_i$ is a random variable and $b_i^* = \inf_D \{b(D)_i\}$. As in Section 3, the density function of $\tilde{\boldsymbol{b}}(D)_i$ is defined over $[b(D)_i - 2s, b(D)_i]$ as $f_D^{(i)}(u) \propto \exp\left(-\frac{\epsilon|u+s-b(D)_i|}{\Delta}\right)$, where $s = \frac{\Delta}{\epsilon} \ln\left(\frac{e^\epsilon - 1}{2\delta} + 1\right)$.

Our mechanism is $(\epsilon m, \delta m)$-DP, a fact that follows immediately from Theorem 3.2 and the composition and post-processing properties of DP [3, 5]. Our quality guarantee depends on the "niceness" of the matrix $\mathbf{A}$, as quantified by the LP's *condition number* [8]: given two norms $||\cdot||_\beta$ and $||\cdot||_\nu$,

$$\gamma_{\beta,\nu}(\mathbf{A}) = \sup\left\{ ||\boldsymbol{u}||_{\beta^*} : \begin{array}{l} ||\mathbf{A}^T\boldsymbol{u}||_{\nu^*} = 1 \\ \text{The rows of } \mathbf{A} \text{ corresponding to nonzero entries of } \boldsymbol{u} \text{ are} \\ \text{linearly independent} \end{array} \right\}.$$

When $\mathbf{A}$ is nonsingular and $||\cdot||_\beta$ and $||\cdot||_\nu$ equal the ℓ_2-norm, $\gamma_{\beta,\nu}(\mathbf{A})$ equals the inverse of the minimum singular value, $\sigma_{\min}(\mathbf{A})^{-1}$. Li [8] proved that $\gamma_{\beta,\nu}(\mathbf{A})$ sharply characterizes the extent to which a change in the constraint scalars $\boldsymbol{b}$ causes a change in the LP's optimal solution.



(a) $\epsilon = 0.25$ (b) $\epsilon = 0.5$ (c) $\epsilon = 0.75$

Figure 2: Comparison of our approach with a baseline based on the Laplace mechanism (Section 5).

Theorem 4.2. *Suppose Assumption 4.1 holds. With probability 1, the linear program in Equation* (2) *is feasible. Let* $\boldsymbol{x}^* \in \mathbb{R}^n$ *be an optimal solution to the original LP (Equation* (1)*). Then* $||\boldsymbol{x}^* - \boldsymbol{x}(D)||_\nu \leq \gamma_{\beta,\nu}(\mathbf{A}) \cdot ||\boldsymbol{b}(D) - \bar{\boldsymbol{b}}(D)||_\beta$.

Proof. We show that $S^* = \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq (b_1^*, \ldots, b_m^*)\}$, which allows us to prove that Equation (2) is feasible (Lemmas C.2, C.3). The bound on $||\boldsymbol{x}^* - \boldsymbol{x}(D)||_\nu$ follows from Theorem 3.3 by Li [8]. □

By definition of $\bar{\boldsymbol{b}}(D)$, for any ℓ_p-norm $||\cdot||_p$, we have that $||\boldsymbol{b}(D) - \bar{\boldsymbol{b}}(D)||_p = \tilde{O}\left(\sqrt[p]{m}\Delta\left(\epsilon^{-1} + 1\right)\right)$. Theorem 4.2 therefore implies that $|\boldsymbol{c}^\top \boldsymbol{x}^* - \boldsymbol{c}^\top \boldsymbol{x}(D)| = \tilde{O}\left(||\boldsymbol{c}||_{\nu^*}\gamma_{p,\nu}(\mathbf{A})\sqrt[p]{m}\Delta\left(\epsilon^{-1} + 1\right)\right)$. When $\mathbf{A}$ is nonsingular, setting $||\cdot||_\beta = ||\cdot||_\nu = ||\cdot||_2$ implies that $|\boldsymbol{c}^\top \boldsymbol{x}^* - \boldsymbol{c}^\top \boldsymbol{x}(D)| = \tilde{O}\left(||\boldsymbol{c}||_2 \sigma_{\min}(\mathbf{A})^{-1}\sqrt{m}\Delta\left(\epsilon^{-1} + 1\right)\right)$.

A natural question is whether we can achieve pure $(\epsilon, 0)$-differential privacy. In Appendix C.1, we prove that if $S^* \neq \emptyset$, then the optimal $(\epsilon, 0)$-DP mechanism disregards the database D and outputs $\text{argmax}_{\boldsymbol{x} \in S^*} \boldsymbol{c}^\top \boldsymbol{x}$ with probability 1. If $S^* = \emptyset$, then no $(\epsilon, 0)$-DP mechanism exists. This shows that any non-trivial private mechanism must allow for a failure probability $\delta > 0$.

5 Experiments

We return now to the single-dimensional LP from Section 3. We compare our approach to the following ϵ-DP baseline: given an offset $t \geq 0$, draw $\eta \sim \text{Laplace}\left(\frac{\Delta}{\epsilon}\right)$, and release $x_{\epsilon,t}(D) := b(D) - t + \eta$. This approach will violate the LP's constraint, but to what extent? We show that when both approaches have equal expected error, $x_{\epsilon,t}(D)$ violates the LP's constraint a non-negligible fraction of the time, whereas our approach never does.

In Figure 2, we plot the offset t along the x-axis. This offset equals the expected error of the mechanism $x_{\epsilon,t}(D)$. For three different values of ϵ, we draw 10,000 samples from Laplace $\left(\frac{1}{\epsilon}\right)$. The blue line equals the fraction of samples η where $\eta > t$. For any such sample and any $b(D)$, we have that $b(D) - t + \eta > b(D)$, which means the LP's constraint is violated. For three different values of δ, we compute the expected error of our mechanism, $\frac{1}{\epsilon}\ln\left(\frac{e^\epsilon - 1}{2\delta} + 1\right)$. We mark this value on the x-axis. Next, we answer the question: if we were to set the offset t to this expected error, how often would the mechanism $x_{\epsilon,t}$ violate the LP's constraint? We mark this value along the y-axis.

As Figure 2 illustrates, the smaller ϵ is, the more advantageous our approach. Intuitively, this is because when ϵ is small, Laplace$\left(\frac{1}{\epsilon}\right)$ is less concentrated, so η will be often be greater than t. Meanwhile, the smaller δ is, the greater the expected error of our mechanism. Therefore, smaller values of δ induce fewer constraint violations when we set the offset t equal to this expected error.

6 Conclusion

We presented a new differentially private method for solving linear programs, where the right-hand side of the constraints $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}$ depends on private data, and where the constraints must always be satisfied. Natural directions for future research would be to allow the matrix $\mathbf{A}$ to also depend on private data, and to generalize the constraints or object function from linear to nonlinear functions.

References

[1] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Differentially private empirical risk minimization: Efficient algorithms and tight error bounds. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, 2014.

[2] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[3] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, 2006.

[4] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Theory of Cryptography Conference (TCC)*, pages 265–284. Springer, 2006.

[5] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[6] Justin Hsu, Aaron Roth, Tim Roughgarden, and Jonathan Ullman. Privately solving linear programs. In *Proceedings of the International Colloquium on Automata, Languages and Programming (ICALP)*, pages 612–624, 2014.

[7] Daniel Kifer, Adam D. Smith, and Abhradeep Thakurta. Private convex optimization for empirical risk minimization with applications to high-dimensional regression. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 25.1–25.40, 2012.

[8] Wu Li. The sharp Lipschitz constants for feasible and optimal solutions of a perturbed linear program. *Linear algebra and its applications*, 187:15–40, 1993.

[9] Olvi L Mangasarian. A condition number of linear inequalities and equalities. *Methods of Operations Research*, 43:3–15, 1981.

[10] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 94–103, 2007.

A Transportation problem linear program

$$\begin{array}{ll} \text{minimize} & \sum_{i,j} c_{ij} x_{ij} \\ \text{such that} & \sum_{j=1}^{N} x_{ij} \leq s_i \quad \wedge \quad \sum_{i=1}^{M} x_{ij} \geq r_j \quad \wedge \quad x_{ij} \geq 0 \quad \forall i \leq M,\ j \leq N \end{array}$$

Figure 3: The transportation problem formulated as a linear program. There are N hospital branches and M pharmacies. Each branch $j \in \{1, \ldots, N\}$ requires $r_j \in \mathbb{R}$ units of a specific drug. Each pharmacy $i \in \{1, \ldots, M\}$ has a supply of $s_i \in \mathbb{R}$ units. It costs $c_{ij} \in \mathbb{R}$ dollars to transport a unit of the drugs from pharmacy i to hospital j. We use the notation x_{ij} to denote the units of drugs transported from pharmacy i to hospital j.

B Omitted proofs from Section 3 about single-dimensional LPs

Via a change of variables, $Z(D) = Z(D')$ for any D, D', so we drop the dependency on D from the normalizing constant Z.

Theorem 3.2. *Our algorithm preserves (ϵ, δ)-DP.*

Proof. Let D and D' be a pair of neighboring datasets and let $V \subseteq \mathbb{R}$ be a set of real values. Without loss of generality, suppose $b(D') \leq b(D)$. Since the support of $x(D)$ equals $[b(D) - 2s, b(D)]$, we

know that

$$\begin{aligned}\Pr[x(D) \in V] &= \Pr[x(D) \in V \cap [b(D) - 2s, b(D')]] + \Pr[x(D) \in V \cap [b(D'), b(D)]] \\ &= \int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du + \int_{b(D')}^{b(D)} f_D(u)\mathbf{1}_{\{u \in V\}}\, du.\end{aligned}$$

Next, since $b(D') \geq b(D) - \Delta$, we know that $\int_{b(D')}^{b(D)} f_D(u)\mathbf{1}_{\{u \in V\}}\, du \leq \int_{b(D)-\Delta}^{b(D)} f_D(u)\mathbf{1}_{\{u \in V\}}\, du$. In Lemma B.1, we prove that $\int_{b(D)-\Delta}^{b(D)} f_D(u)\mathbf{1}_{\{u \in V\}}\, du \leq \delta$, which means that

$$\Pr[x(D) \in V] \leq \int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du + \delta. \tag{3}$$

We now bound the first summand of Equation (3)'s right-hand-side, $\int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du$, by $e^\epsilon \Pr[x(D') \in V]$. By definition of f_D,

$$\begin{aligned}\int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du &= \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon|u + s - b(D)|}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\ &= \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon|u + s - b(D') + b(D') - b(D)|}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du.\end{aligned}$$

By the reverse triangle inequality, $\int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du$ is upper-bounded by

$$\frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon(|u + s - b(D')| - |b(D') - b(D)|)}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du.$$

Since D and D' are neighboring, $|b(D') - b(D)| \leq \Delta$, which means that

$$\begin{aligned}\int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du &\leq \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon(|u + s - b(D')| - \Delta)}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\ &= \frac{e^\epsilon}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon|u + s - b(D')|}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\ &= e^\epsilon \int_{b(D)-2s}^{b(D')} f_{D'}(u)\mathbf{1}_{\{u \in V\}}\, du \\ &\leq e^\epsilon \Pr[x(D') \in V].\end{aligned}$$

This inequality together with Equation (3) implies that $\Pr[x(D) \in V] \leq e^\epsilon \Pr[x(D') \in V] + \delta$.

By a similar argument, we show that $\Pr[x(D') \in V] \leq e^\epsilon \Pr[x(D) \in V] + \delta$:

$$\begin{aligned}
& \Pr[x(D') \in V] \\
= \; & \Pr[x(D') \in V \cap [b(D') - 2s, b(D) - 2s]] + \Pr[x(D') \in V \cap [b(D) - 2s, b(D')]] \\
= \; & \int_{b(D')-2s}^{b(D)-2s} f_{D'}(u)\mathbf{1}_{\{u \in V\}}\, du + \int_{b(D)-2s}^{b(D')} f_{D'}(u)\mathbf{1}_{\{u \in V\}}\, du \\
\leq \; & \int_{b(D')-2s}^{b(D')-2s+\Delta} f_{D'}(u)\mathbf{1}_{\{u \in V\}}\, du + \int_{b(D)-2s}^{b(D')} f_{D'}(u)\mathbf{1}_{\{u \in V\}}\, du \\
\leq \; & \delta + \int_{b(D)-2s}^{b(D')} f_{D'}(u)\mathbf{1}_{\{u \in V\}}\, du \\
= \; & \delta + \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon|u+s-b(D')|}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\
= \; & \delta + \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon|u+s-b(D)+b(D)-b(D')|}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\
\leq \; & \delta + \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon(|u+s-b(D)|-|b(D)-b(D')|)}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\
\leq \; & \delta + \frac{1}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon(|u+s-b(D)|-\Delta)}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\
= \; & \delta + \frac{e^\epsilon}{Z}\int_{b(D)-2s}^{b(D')} \exp\left(-\frac{\epsilon|u+s-b(D)|}{\Delta}\right)\mathbf{1}_{\{u \in V\}}\, du \\
= \; & \delta + e^\epsilon \int_{b(D)-2s}^{b(D')} f_D(u)\mathbf{1}_{\{u \in V\}}\, du \\
\leq \; & e^\epsilon \Pr[x(D) \in V] + \delta.
\end{aligned}$$

We conclude that for any pair of neighboring databases D and D', $\Pr[x(D') \in V] \leq e^\epsilon \Pr[x(D) \in V] + \delta$, so differential privacy holds. □

We now prove that the distribution defined by f_D has tails with probability mass bounded by δ, a fact that we use in the privacy guarantee above.

Lemma B.1. *The probability mass of $x(D)$ in each of the intervals $[b(D) - 2s, b(D) - 2s + \Delta]$ and $[b(D) - \Delta, b(D)]$ is δ. In other words,*

$$\int_{b(D)-2s}^{b(D)-2s+\Delta} f_D(u)\, du = \int_{b(D)-\Delta}^{b(D)} f_D(u)\, du = \delta.$$

Proof. Since the density function f_D is symmetric around $b(D) - s$, we prove this lemma by proving that $\int_{b(D)-\Delta}^{b(D)} f_D(u)\, du = \delta$. By definition,

$$\int_{b(D)-\Delta}^{b(D)} f_D(u)\, du = \frac{1}{Z}\int_{b(D)-\Delta}^{b(D)} \exp\left(-\frac{\epsilon|u+s-b(D)|}{\Delta}\right) du.$$

Since $b(D) - \Delta \geq b(D) - s$ we can remove the absolute value from this expression:

$$\int_{b(D)-\Delta}^{b(D)} f_D(u)\, du = \frac{1}{Z}\int_{b(D)-\Delta}^{b(D)} \exp\left(-\frac{\epsilon(u+s-b(D))}{\Delta}\right) du = \frac{\Delta\left(e^\epsilon - 1\right)e^{-s\epsilon/\Delta}}{Z\epsilon}.$$

Since $Z = \frac{2\Delta\left(1-e^{-\epsilon s/\Delta}\right)}{\epsilon}$, we have that

$$\int_{b(D)-\Delta}^{b(D)} f_D(u)\, du = \frac{\left(e^\epsilon - 1\right)e^{-s\epsilon/\Delta}}{2\left(1-e^{-\epsilon s/\Delta}\right)} = \frac{e^\epsilon - 1}{2\left(e^{s\epsilon/\Delta} - 1\right)} = \delta,$$

as claimed. □

C Omitted proofs from Section 4 about multi-dimensional LPs

Proposition C.1. *Suppose D and D' are two neighboring databases with disjoint feasible regions: $\{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\} \cap \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D')\} = \emptyset$. There is no (ϵ, δ)-DP mechanism with $\delta < 1$ that satisfies the constraints with probability 1.*

Proof. For the sake of a contradiction, suppose $\mu : 2^{\mathcal{X}} \to \mathbb{R}^n$ is an (ϵ, δ)-DP mechanism with $\delta < 1$ that satisfies the constraints with probability 1. Let $V = \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$. Since $V \cap \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D')\} = \emptyset$, it must be that $\Pr[\mu(D') \in V] = 0$. This means that $1 = \Pr[\mu(D) \in V] \leq e^{\epsilon} \Pr[\mu(D') \in V] + \delta = \delta$, which is a contradiction. Therefore, the lemma statement holds. □

Lemma C.2. *With probability 1, the linear program in Equation* (2) *is feasible.*

Proof. By definition, the constraint vector $\bar{\boldsymbol{b}}$ is component-wise greater than the vector $\boldsymbol{b}^* = (b_1^*, \ldots, b_m^*)$, where $b_i^* = \inf_{D \subseteq \mathcal{X}} b(D)_i$. Therefore, $\left\{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \bar{\boldsymbol{b}}(D)\right\} \supseteq \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*\}$. By Lemma C.3, we know that $\{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*\} = \bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$ and by Assumption 4.1, we know that $\bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$ is nonempty. Therefore, the feasible set of the linear program in Equation (2), $\left\{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \bar{\boldsymbol{b}}(D)\right\}$, is nonempty. □

We now prove Lemma C.3, which we used in the proof of Lemma C.2. Lemma C.3 guarantees that the (nonempty) intersection of the feasible regions across all databases is equal to the set of all $\boldsymbol{x}$ such that $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*$.

Lemma C.3. *The set $\bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$ is equal to the set $\{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*\}$.*

Proof. Suppose that $\boldsymbol{x} \in \bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$. We claim that $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*$. To see why, let $\boldsymbol{a}_i$ be the i^{th} row of the matrix $\mathbf{A}$. For a contradiction, suppose that for some row $i \in [m]$, $\boldsymbol{a}_i^\top \boldsymbol{x} > b_i^*$, and let $\gamma = \boldsymbol{a}_i^\top \boldsymbol{x} - b_i^*$. Since $b_i^* = \inf_{D \subseteq \mathcal{X}} b(D)_i$, there exists a database D such that $b(D)_i < b_i^* + \frac{\gamma}{2}$. Since $\boldsymbol{x} \in \bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$, it must be that $\boldsymbol{a}_i^\top \boldsymbol{x} \leq \boldsymbol{b}(D)_i < b_i^* + \frac{\gamma}{2} = \frac{1}{2}\left(\boldsymbol{a}_i^\top \boldsymbol{x} + b_i^*\right)$. This chain of inequalities implies that $\boldsymbol{a}_i^\top \boldsymbol{x} < b_i^*$, which is a contradiction. Therefore, $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*$.

Next, suppose $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*$. Then $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)$ for every database D, which means that $\boldsymbol{x} \in \bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$. We conclude that $\bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\} = \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}^*\}$. □

C.1 Characterization of $(\epsilon, 0)$-differentially private mechanisms

We conclude with a complete characterization of $(\epsilon, 0)$-differentially private mechanisms.

Theorem C.4. *Let $S^* = \bigcap_{D \subseteq \mathcal{X}} \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}(D)\}$ be the intersection of all feasible sets across all databases D. If S^* is nonempty, then the optimal $(\epsilon, 0)$-differentially private mechanism outputs $\operatorname{argmax}_{\boldsymbol{x} \in S^*} \boldsymbol{c}^\top \boldsymbol{x}$ with probability 1. If S^* is empty, then no $(\epsilon, 0)$-differentially private mechanism exists.*

Proof. Fix a mechanism, and let $P(D)$ be the set of vectors $\boldsymbol{x}$ in the support of the mechanism's output given as input the database D. We claim that if the mechanism is $(\epsilon, 0)$-differentially private, then there exists a set P^* such that $P(D) = P^*$ for all databases D. Suppose, for the sake of a contradiction, that there exist databases D and D' such that $P(D) \neq P(D')$. Let $D_1, \ldots, D_n$ be a sequence of databases such that $D_1 = D$, $D_n = D'$, and each pair of databases D_i and D_{i+1} are neighbors. Then there must exist a pair of neighboring databases D_i and D_{i+1} such that $P(D_i) \neq P(D_{i+1})$, which contradicts the fact that the mechanism is $(\epsilon, 0)$-differentially private. Therefore, if the mechanism is $(\epsilon, 0)$-differentially private, then to satisfy the feasibility requirement, we must have that $P^* \subseteq S^*$. If S^* is empty, then no such mechanism exists. If S^* is nonempty, then the optimal $(\epsilon, 0)$-differentially private mechanism outputs $\operatorname{argmax}_{\boldsymbol{x} \in S^*} \boldsymbol{c}^\top \boldsymbol{x}$ with probability 1. □